

Gödel Numbering

Background

Kurt Gödel is considered to be one of the most significant logicians in history. His various mathematical discoveries and proofs had a profound impact on the field of computer science, specifically computability theory [\[Gödel\]](#). One of his proof techniques involved a method of mapping a sequence of symbols to a unique integer number [\[Gödel Numbering\]](#). In this assignment you will implement the Gödel Numbering system.

The Problem: Gödel Numbering

Write a program to implement Gödel Numbering as follows:

Given a sequence of n positive integers $(X_1, X_2, X_3, \dots, X_n)$, compute an integer e using the formula

$$e = 2^{x_1} \times 3^{x_2} \times 5^{x_3} \times \dots \times p_n^{x_n} = \prod_{i=1}^n p_i^{x_i}$$

where $2, 3, 5, \dots, p$ are sequential prime numbers.

Your program must compute the ascending sequence of prime numbers. It must not have the prime numbers stored in the source code, read from a file, or utilize any pre-computed values.

Note that to be effective, the Gödel Numbering system must be invertible. That is, the mapping

$$(x_1, x_2, x_3, \dots, x_n) \rightarrow e$$

must be able to be reversed as such

$$e \rightarrow (x_1, x_2, x_3, \dots, x_n)$$

This reverse mapping is achieved by finding the prime factors (in ascending order) of e and counting the number of times each factor occurs.

Example:

$$\{1, 2, 3\} \rightarrow 2250 \rightarrow \{2, 3, 3, 5, 5, 5\} \rightarrow \{1, 2, 3\}$$

where $\{2, 3, 3, 5, 5, 5\}$ are the prime factors of 2250.

You will need to provide code to

1. Compute a sequence of prime numbers

2. Compute the set of prime factors of a composite number
3. Perform the Gödel Numbering algorithms (forward and reverse)

I highly recommend that you create methods or classes to perform these operations.

The value of e gets large very quickly as the input sequence lengthens. Therefore, you should use Java's **long** integer data type (avoid **byte**, **short** and **int** integers.) The test case I give you below will compute correct values using **long** Integer types.

Deliverables

1. Your source code (.java) files for each class.
2. A reflective essay describing
 - a. Successes
 - b. Difficulties and how you overcame them
 - c. How you tested your code to ensure correctness
 - d. A screen shot of your program processing the input sequence {7, 6, 5, 4, 3, 2, 1}.

Your program must compute the forward mapped value e and the reverse mapped sequence from e . The reverse mapped sequence should be the same as the original input sequence. Display the results on a Java console.